Classify the Emotions of Each Sentences Using Reliable Features

Haiyue Sun 2100013127@stu.pku.edu.cn

Abstract

Sexism is a growing problem on the internet. To solve the problem of classifying the emotion of sentences, I proposed two types of models. One is based on Bertblock. The other is Log- linear, which uses One-hot to obtain features. After comparing the performance of these two models, it was found that the Log-linear model has a better performance than the Bertblock+Linear model. Considering of this abnormal situation, some assumptions were also given. My code is in this github linkgithub

1 Introduction

As sexism is a growing problem on the Internet, the explicable detection of online sexism is becoming a new challenge. This problem is a concrete example of an emotion classification task. The task is to classify the given sentences into the given types.(Hannah Rose Kirk, 2023) During this time, models based on Transformer, such as GPT-2(Radford, 2019b) and GoogleBert(Radford, 2019a), perform better than before. However, the training of a Transformer-based model without sufficient power is difficult. Inspired by transfer learning, I use pre-trained models, such as bert-based-uncased and bert-large-uncased(Jacob Devlin, 2018), with a linear classifier to achieve the goal. I program a loglinear model using the one-hot feature extraction method to better demonstrate the difference of this model.

In this article, I will present two models, the Bert-base+Fc model and the Loglinear model. After discussing the structure of these models, I give the results of both. I also try to give reasonable explanations for the result.

2 Method

2.1 Problem modeling

The goal of this problem is to find a function that can correctly classify arbitrary sentences into given types. The i-th sentence is assumed to be a list of words and its form is $X_i = \{x_1, x_2, ..., x_k\}$, where k is the length of the sentence. Each sentences has a label $y_i \in Y$. To find the most correct label of an unlabelled sentence, a model $f(\hat{y_{ij}}|X_i)$ is needed. The output of the function should be a probability, so the result of the function is below:

$$\hat{y_i} = \arg\max_{y_{ij} \in Y} f(\hat{y_{ij}}|X_i)$$

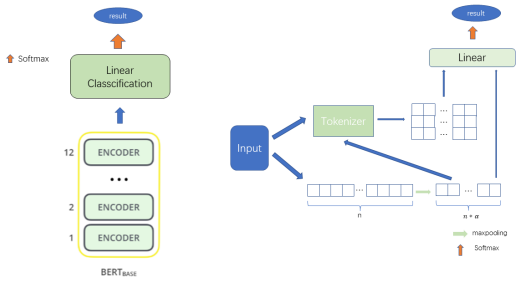

Figure 1: Bert+Fc and Loglinear models. The left one is the structure of Bert+Fc model. The right one is Loglinear model.

2.2 Loss Function

It is natural for me to use the Cross-Entropy function for the binary classification problem.

$$CrossEntropy = -y_0 log p_1 - y_1 log p_0$$

As for the multi-classification problem, my idea is to train k models that can classify one type from others. And the result is to get the category with the highest probability. So using the Cross-Entropy function as the loss function is a matter of course.

2.3 Bert+Fc Model

Using a pre-trained Bert-base-uncased model with a linear classifier, it is much easier to build a model based on Transformer. (Figure 1) In this model,

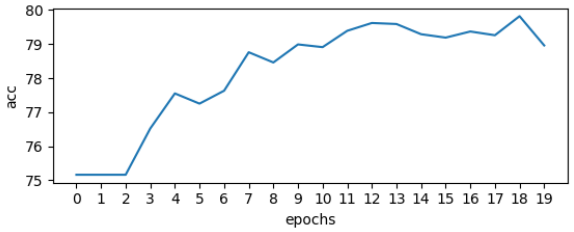

Figure 2: Bert+Fc models accuracy performed on test set.

I assume that sentences share features directly related to labels. Therefore, the first step in the process is to extract features from the sentences. Taking advantage of the attention mechanism in Transformer, it is much easier to obtain global features from the sentences. In this model, this is the goal of the Bert block. After obtaining the features, a weighted sum over the given features is performed. The result is the output of the softmax function.

The difficulty in training this model is that the Berttokenizer's dictionary can't encode all the symbols in this task. Some symbols like ":(" or emojis are not typical text information but they also express a certain emotion of the user. These "special" symbols often play an important role in the construction of an ironic sentence. In my model I just ignore them and always keep text information. However, I think we can give them special signals or construct a graph containing these symbols and some words with similar meanings, so we can represent them using other available words.

2.4 Loglinear Model

For this model, I assume that each word in a sentence has nothing to do with each other and the feature of the sentence is just a combination of the number of each word in our dictionary that appears in the sentence. (Figure 1) Then I can do a weighted sum over the given features and use the softmax function to get the final results.

Incidentally, to compare performance with the Bert+fc model, the Loglinear model is trained on the same dataset.

3 Result

The training result exceeds my expectations. The performance of the loglinear model is better than that of the Bert+fc model, with 80% accuracy compared to 79.8% accuracy.(Figure 2,3) Besides, the Bert+fc model also performs worse than the loglinear model (80% accuracy on average compared to

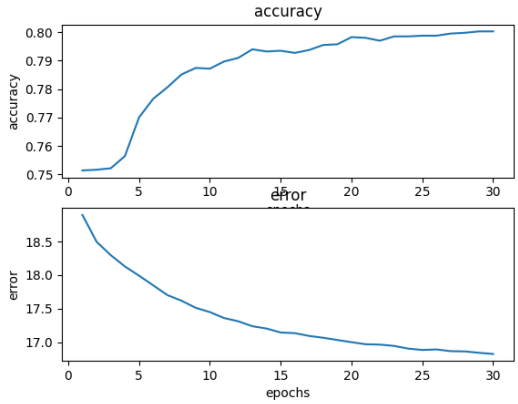

Figure 3: Loglinear models accuracy performed on test set.

90% accuracy).

For the first problem, I think there are two main reasons. It is because the dimension of the output features of the Bert block (768) is much lower than the One-Hot method (>3000), which makes it difficult to classify so many sentences. The other reason is that the Berttokenizer is not trained for this specific task, which will lead to some mismatching problems.

As for the second problem, the probable reason from my perspective is that, benefiting from the attention mechanism in Transformer, Bert is not easy to overfit.

4 Conclusion

In this homework I presented two basic models that try to deal with explicable detection of online sexism task. It was found that if we build models based on Transformer, it is much easier to overcome the overfitting problem. The dimension of the features plays a pivotal role in classification. The key part of the emotion classification task may lie in how to extract more and more meaningful features and how to identify the relationship between these features.

References

Bertie Vidgen Paul Röttger Hannah Rose Kirk, Wenjie Yin. 2023. Semeval-2023 task 10: Explainable detection of online sexism. arXiv:2303.04222.

Kenton Lee Kristina Toutanova Jacob Devlin, Ming-Wei Chang. 2018. Bert: Pre-training of deep bidi-

rectional transformers for language understanding. arXiv:1810.04805.

Alec Radford. 2019a. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805v2.

Alec Radford. 2019b. Language models are unsupervised multitask learners.